



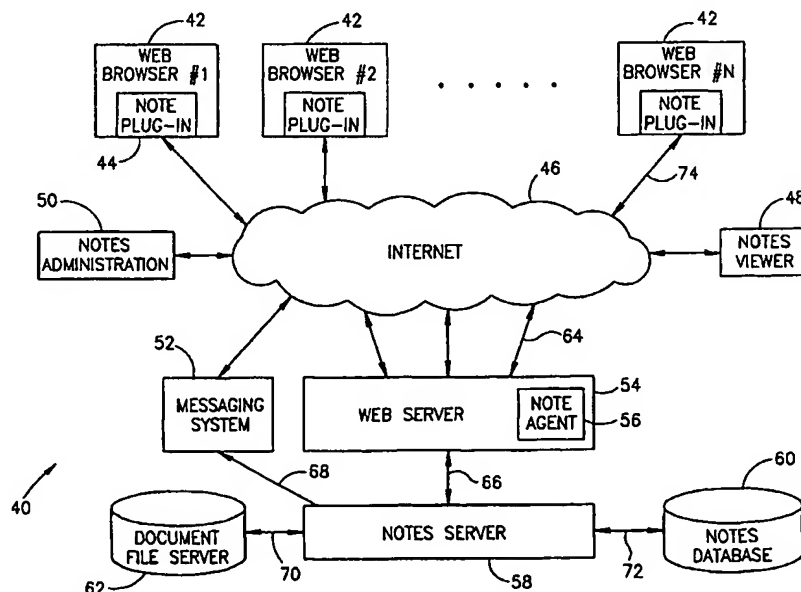
INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

| | | |
|--|--|--|
| (51) International Patent Classification ⁷ : H04N | A2 | (11) International Publication Number: WO 00/56055 (43) International Publication Date: 21 September 2000 (21.09.00) |
| <p>(21) International Application Number: PCT/IL00/00155</p> <p>(22) International Filing Date: 14 March 2000 (14.03.00)</p> <p>(30) Priority Data: 09/270,255 15 March 1999 (15.03.99) US</p> <p>(71) Applicant: RTIMAGE LTD. [IL/IL]; Hamasger Street 6, 60395 Or Yehuda (IL).</p> <p>(72) Inventors: EINTRACHT, Zvika; Hayarden Street 27, 52281 Ramat Gan (IL). OVSIANKIN, Alexander; Dakar Street 3, 71351 Lod (IL). DEMENTIEV, Roman; Sheinkin Street 50, 53304 Givatayim (IL). SIDEMAN, Gil; Pardes Meshutaf Street 56, 43356 Raanana (IL).</p> <p>(74) Agent: ZARETSKY, Howard; P.O. Box 12735, 46733 Herzelia (IL).</p> | <p>(81) Designated States: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).</p> <p>Published Without international search report and to be republished upon receipt of that report.</p> | |

(54) Title: COLLABORATIVE DOCUMENT ANNOTATION SYSTEM

(57) Abstract

A system for collaborative document annotation whereby notes or annotations associated with a document, such as an image or text document, are stored on a web server. The documents and associated annotations are treated independently from each other whereby separate data structures are created for the documents and for the associated annotations. A web server application on the server side functions to capture special requests from one or more client applications for creating, storing, editing and retrieving annotations related to specific documents located in the server. A Notes Server functions to log all annotation activities along with information about the corresponding clients that create, edit and retrieve them. On the client side a Notes Client software application, that is implementable as a web browser Plug-In, functions to display the document that the user wishes to annotate and provides the tools necessary to permit the user to create, edit, retrieve and store notes. A synchronization process transmits the annotations generated by the user from the client to the server. In response, the server transmits back an acknowledgement along with any new notes that other clients may have posted since the last synchronization was performed. Multiple clients can annotate a document in an asynchronous fashion wherein the document resides on a central web server. When annotations are posted to the server by a client, the state of the annotation database is synchronized such that all other clients can retrieve the current, up to date annotations associated with a document.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

| | | | | | | | |
|----|--------------------------|----|--|----|--|----|--------------------------|
| AL | Albania | ES | Spain | LS | Lesotho | SI | Slovenia |
| AM | Armenia | FI | Finland | LT | Lithuania | SK | Slovakia |
| AT | Austria | FR | France | LU | Luxembourg | SN | Senegal |
| AU | Australia | GA | Gabon | LV | Latvia | SZ | Swaziland |
| AZ | Azerbaijan | GB | United Kingdom | MC | Monaco | TD | Chad |
| BA | Bosnia and Herzegovina | GE | Georgia | MD | Republic of Moldova | TG | Togo |
| BB | Barbados | GH | Ghana | MG | Madagascar | TJ | Tajikistan |
| BE | Belgium | GN | Guinea | MK | The former Yugoslav Republic of Macedonia | TM | Turkmenistan |
| BF | Burkina Faso | GR | Greece | | | TR | Turkey |
| BG | Bulgaria | HU | Hungary | ML | Mali | TT | Trinidad and Tobago |
| BJ | Benin | IE | Ireland | MN | Mongolia | UA | Ukraine |
| BR | Brazil | IL | Israel | MR | Mauritania | UG | Uganda |
| BY | Belarus | IS | Iceland | MW | Malawi | US | United States of America |
| CA | Canada | IT | Italy | MX | Mexico | UZ | Uzbekistan |
| CF | Central African Republic | JP | Japan | NE | Niger | VN | Viet Nam |
| CG | Congo | KE | Kenya | NL | Netherlands | YU | Yugoslavia |
| CH | Switzerland | KG | Kyrgyzstan | NO | Norway | ZW | Zimbabwe |
| CI | Côte d'Ivoire | KP | Democratic People's Republic of Korea | NZ | New Zealand | | |
| CM | Cameroon | | | PL | Poland | | |
| CN | China | KR | Republic of Korea | PT | Portugal | | |
| CU | Cuba | KZ | Kazakhstan | RO | Romania | | |
| CZ | Czech Republic | LC | Saint Lucia | RU | Russian Federation | | |
| DE | Germany | LI | Liechtenstein | SD | Sudan | | |
| DK | Denmark | LK | Sri Lanka | SE | Sweden | | |
| EE | Estonia | LR | Liberia | SG | Singapore | | |

COLLABORATIVE DOCUMENT ANNOTATION SYSTEM

FIELD OF THE INVENTION

The present invention relates generally to collaborative work systems and more particularly relates to a system for collaborative document annotation based on the exchange of notes attached to web based documents, the system being suitable for use over a network such as the Internet based World Wide Web (WWW).

BACKGROUND OF THE INVENTION

A basic function of World Wide Web browsers in common use today, such as Netscape Navigator and Microsoft Internet Explorer, is to allow web pages stored in remote Web server to be retrieved and viewed. The user initiates a viewing session on a web browser by specifying a particular URL corresponding to a page to be viewed. The web server transmits the various elements of the requested page to the browsing application. Any special data formats, e.g., video, audio, certain images, documents not written in HTML language, etc., are treated by one or more web browser software plug-ins on the client computer.

A common need is to be able to view the document, i.e., images, etc., on the client computer and be able to add one or more annotations to the document. It is becoming more and more common for groups of two or more people to work collaboratively on a project. The members of the group may or may not be located in the same geographic area. The collaboration is often via a network such as the global Internet. The members of the group may be located anywhere in the world as long they have a connection to the Internet.

The ability to annotate documents displayed on screen is very useful when information needs to be exchanged among more than one person. It would be beneficial if one person's comments could be shared among many others by attaching notes that travel with the underlying document being worked on.

Currently, however, it is not possible to annotate documents on the client side such that other clients can see the notes. The entire document with the attached notes must be transmitted to other clients for them to see the notes. Transmitting the entire document from client to client through the server in order to exchange annotations is burdensome, inefficient and costly in terms of time and bandwidth.

The majority of prior art systems that deal with image and page annotation are adapted to handle annotation only on the client side. In these systems, a user launches an editing tool to open a document locally on her/his computer. The document is edited locally with the other clients not being to easily view the changes or notes. Some of these
5 editing tools permit attaching meta information such as annotations to the document. These systems operate by saving the annotations within the local file system that is also used to store the document.

Also known in the prior art are systems that permit collaboration that is based on the replication of the display at one client system to the display on another client system.
10 These systems, however, require that all parties to the collaboration session be logged on to the server at the same time in order to view the annotations made by one or more of the parties.

Thus, there is strong felt need for a system that would allow multiparty collaboration based on the asynchronous exchange of annotations over a network such as
15 the Internet without the requirement that all parties wishing to collaborate be simultaneously logged on to a server.

SUMMARY OF THE INVENTION

The present invention is a system for collaborative document annotation whereby notes or annotations associated with a document, such as an image or text document, are stored on a web server. The documents and associated annotations are treated
5 independently from each other. Separate data structures are created for the documents and for the associated annotations thus permitting their independent management.

The invention can be implemented as software, a portion of which executes on the server side and a portion executes on the client side. The server side may comprise a plurality of software applications running in parallel that in combination provide the server
10 functionality of the invention. A web server application on the server side functions to capture special requests from one or more client applications for creating, storing, editing and retrieving annotations related to specific documents located in the server. A notes server functions to log all annotation activities along with information about the corresponding clients that create, edit and retrieve them.

15 The notes information are transmitted between client and server applications via TCP/IP protocols over communications means such as a LAN, WAN or the Internet. The invention may be adapted to operate over an Intranet, e.g., LAN, or Extranet, e.g., WAN or the Internet.

On the client side a client software application functions to display the document
20 that the user wishes to annotate and provides the tools necessary to permit the user to create, edit, retrieve and store notes. Note that the client software application can be implemented as a web browser plug-in module. The plug-in contains the user interface for navigating within the document and for handling the notes.

A synchronization button is provided which, when pressed by the user, transmits
25 the annotations generated by the user from the client to the server using a particular protocol. In response, the server transmits back an acknowledgement along with any new notes that other clients may have posted since the last synchronization was performed.

A user may annotate many documents at the same time by opening several web browser windows. In addition, other clients can annotate either the same document or
30 other documents at the same or a later time. In accordance with the invention, the annotations are transmitted from the server independent of the data transmitted that is

related to the viewed document. At the client side, the client application layers the annotations over the image (or document) in accordance with the coordinates of each.

As result, the present invention has the advantage of permitting multiple clients to annotate a document that resides on a central web server in an asynchronous fashion.

5 When annotations are posted to the server by a client, the state of the annotation database is synchronized such that all other clients can retrieve the current, up to date annotations associated with a document. As an option, the server can be adapted to trigger one or more alarms in response to a change being made to the database maintained by the server. Those clients that wish to be notified in the event of a database change at the server, can be
10 notified by means of visual indications, audio and/or e-mail. In addition, the alarms can be filtered in numerous ways. Examples of filtering include, but are not limited to, specific documents, document types, document folders, annotation contents and specific clients.

The system of the present invention further allows annotations to be kept private. Annotations can be kept secure from changes made by non authorized clients. The time an
15 annotation was created is also tracked and stored in terms of the clock in the server. When the time stamp of the annotation is shown on the client, it is shown in terms of the clock on the client.

There is provided in accordance with the present invention a system for annotating documents comprising a document file located on a server, the document file for storing
20 one or more documents, a notes database located on the server, the notes database for storing one or more notes, each note associated with a particular document, one or more notes clients coupled to a network, each notes client operative to locally display a representation of a document remotely stored on the server in the document file, the notes client adapted to permit a user to annotate the document with one or more notes, the notes
25 client operative to simultaneously display the one or more notes associated with the document over the displayed document such that the document is viewable along with the one or more notes, a notes server coupled to the network, the notes server operative to store the documents in the document file separately from notes stored in the notes database, the notes server receiving one or more notes associated with a particular document from the
30 one or more notes clients and synchronization means within the notes clients and the notes server, the synchronization means for updating the notes server with any notes events processed by the notes clients and for updating the notes client with the results of

synchronization updates previously performed by other notes clients since the last synchronization event.

The notes database comprises a Notes Table for storing the current content of a note and a Notes Log Table for storing a transaction history of all note events for notes associated with a particular document. The notes client comprises a display, network
5 communication means, a processor suitably programmed to run a web browser and a client note plug-in adapted to run within the web browser.

The network comprises the Internet, a TCP/IP network or an Intranet. The notes server comprises network communication means, a processor suitably programmed to
10 provide web server services, provide notes agent services, provide document and notes browsing and synchronization services to the one or more notes clients. The notes agent services comprises an Internet Server Application Programming Interface (ISAPI).

The system further comprises a notes administrator for providing one or more Universal Resource Location (URL) aliases in managing and administrating the notes
15 server and a note viewer for generating reports on the history of annotations on a per user or a per document basis that have occurred over a period of time.

The system permits two or more user to collaboratively work on the same document whereby each user is capable of viewing annotations made by the other users upon
20 synchronization with the notes server. A single page of a multipage document can be annotated on a notes client without requiring the entire multipage document to be transmitted to the notes client.

The system further comprises security means for providing Universal Resource Locator (URL) translation and redirection services so as to provide user level security for documents and notes stored on the server. The security means comprises a Document
25 Location Table and a Client Access Key Table.

The document comprises an image or a document consisting of a page description language. The notes client is adapted to permit the user to annotate the document in parallel with the display of the document. The document comprises a plurality of pages whereby each page can be annotated independently of all the other pages. The
30 synchronization means is user initiated or is triggered by the notes server via the signaling of selected notes clients.

There is also provided in accordance with the present invention a method of annotating documents, the method comprising the steps of getting a first request

comprising a Universal Resource Locator (URL) associated with a document to be viewed and annotated and transmitting the first request from a client to a server, sending a first response from the server in response to the first request, the first response comprising the document type of and a representation of the requested document, getting a second request
5 comprising a document note URL corresponding to the notes associated with the document and sending the second request from the client to the server, sending a second response from the server in response to the second request, the second response comprising one or more notes associated with the document, displaying the representation of the document and the associated one or more notes locally on the client such that both are viewable
10 simultaneously, permitting the user to perform annotation functions on the client so as to generate a third request comprising one or more note events related to the document that are transmitted from the client to the server and sending a third response from the server to the client in response to the third request, the third response comprising an updated list of notes associated with the document, the list of notes incorporating any note events
15 previously received from other clients.

The step of getting a first request comprises the step of querying the user for a user name and password. The step of sending a first response comprises the step of authenticating the user on the server before sending the first response. The method further comprises the step of getting a fourth request comprising a folder URL associated with the
20 folder comprising one or more documents to be viewed and annotated and transmitting the fourth request from the client to the server.

The method further comprises the step of sending a fourth response from the server in response to receiving a request comprising a folder URL, the fourth response comprising a directory listing of the folder. The document comprises an image or a document
25 represented using a page description language.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention is herein described, by way of example only, with reference to the accompanying drawings, wherein:

5 Figs. 1A, 1B, 1C are illustrations of an example image displayed with and without multiple annotation displayed thereon;

Fig. 2 is an illustration of an example annotation;

Fig. 3 is a block diagram illustrating the collaborative document annotation system of the present invention;

10 Fig. 4 is a flow diagram illustrating the initialization portion of the collaborative document annotation system of the present invention;

Fig. 5 is a message flow diagram illustrating the transaction sequence between client and server during initialization;

Fig. 6 is a flow diagram illustrating the document browsing portion of the collaborative document annotation system of the present invention;

15 Fig. 7 is a message flow diagram illustrating the transaction sequence between client and server during document browsing;

Fig. 8 is a flow diagram illustrating the note synchronization portion of the collaborative document annotation system of the present invention;

20 Fig. 9 is a message flow diagram illustrating the transaction sequence between client and server during note synchronization;

Fig. 10 is a diagram illustrating the Client Annotation Event Data Structure in more detail;

Fig. 11 is a diagram illustrating the note data structure of the Client Annotation Event Data Structure of Figure 10 in more detail;

25 Fig. 12 is a diagram illustrating the Server Annotation Response Data Structure in more detail; and

Fig. 13 is a diagram illustrating the note data structure of the Server Annotation Response Data Structure of Figure 12 in more detail.

DETAILED DESCRIPTION OF THE INVENTION

Notation Used Throughout

The following notation is used throughout this document.

| Term | Definition |
|-------|---|
| GMT | Greenwich Mean Time |
| HTML | Hypertext Text Markup Language |
| HTTP | Hypertext Transport Protocol |
| IIS | Internet Information Server |
| IP | Internet Protocol |
| ISAPI | Internet Server Application Programming Interface |
| LAN | Local Area Network |
| MIME | Multipurpose Internet Mail Extension |
| NFS | Network File System |
| PDF | Portable Document Format |
| RAM | Random Access Memory |
| RGB | Red Green Blue |
| SMTP | Simple Mail Transport Protocol |
| SQL | Structured Query Language |
| TCP | Transport Control Protocol |
| TIFF | Tag Image File Format |
| URL | Universal Resource Locator |
| WAN | Wide Area Network |
| WWW | World Wide Web |

The following terminology and definitions apply throughout this document.

| Term | Definition |
|----------------------|---|
| Annotation (Note) | A portion of text or a graphical drawing that is associated with a specific location in a document. |
| Annotation Session | The time period during which a user creates, edits, retrieves and stores one or more notes associated with a document. |
| Note Alarms | Notifications that the Notes Server received a Note Event. |
| Note Anchor | The specific location in a document associated with a note. |
| Note Client | The application that displays the annotated document and allows for annotation sessions. |
| Note Contents | The text or graphical information contained in the note. |
| Note Document ID | Identifier for documents and sets of aggregated documents organized under the same logical structure. |
| Note Event | A message related to a note that did not exist at the time of the last synchronization with the Notes Server, or a note that was modified or deleted since that time. |
| Note Owner ID | Identifier used to differentiate between notes generated by various users. |
| Note Persistence | Mechanism whereby the Notes Clients saves the Note Events to the local file system, allowing for synchronization at a later time. |
| Note Serial Number | A number uniquely identifying a note. |
| Note Server | The application that provides the centralized role of document and note management. |
| Note Synchronization | The update activity whereby one or more Note Events are exchanged between the Notes Server and the Notes Client. |
| Note Time Stamp | The time stamp saved on the Notes Server, stored using time zone of the Notes Server. |
| Notes Database | The database maintained by the Notes Server holding all note related information. |
| Notes Log | A transactional history of Notes Events. |

General Description

- 5 The present invention is a system for collaborative document annotation whereby notes or annotations associated with a document are stored on a web server. Examples of documents include, but are not limited to, images, text documents or documents expressed in a page description language such as Postscript or Adobe PDF. In addition, each

document may contain more than one page, wherein each page is annotated independently of the other. The documents and associated annotations are treated independently from each other. Separate data structures are created for the documents and for the associated annotations thus permitting their independent management. The invention can be
5 implemented as software, a portion of which executes on the server side and a portion executes on the client side. The server side may comprise a plurality of software applications running in parallel that in combination provide the server functionality of the invention. A web server application on the server side functions to capture special requests from one or more client applications for creating, storing, editing and retrieving
10 annotations related to specific documents located in the server. A notes server functions to log all annotation activities along with information about the corresponding clients that create, edit and retrieve them.

Illustrations of an example image displayed with and without multiple annotation displayed thereon are shown in Figures 1A, 1B and 1C. The system of the present
15 invention is suitable for use with a variety of document types, including text and image based documents. For illustrative purposes, the following description uses an image type document to illustrate the principles of the invention. This is not meant to limit the scope of the invention. One skilled in the computer and software art can apply the system of the invention to other document types as well.

20 In Figure 1A, the display 10 is shown comprising a single document window 12 in which an image is displayed. No annotations are displayed in this view. In Figure 1B, the display 10 is shown comprising a single document window 14 in which an image is displayed with a plurality of annotations 16. The annotations 16 are displayed over the image but are not a part of the image itself. In Figure 1C, the display 10 is shown
25 comprising two portions: the image window 18 and a Note List window 20. The image window 18 displays the image with the plurality of annotations 16 on top of the image. The Note List window 20 displays a list of all the annotations associated with the image shown in window 18. The user can optionally sort the Note List by Note Owner ID, Note Time Stamp, Note color label and Note Contents. The display may include Note Event
30 description, Note Owner ID, user name, path, originator IP address, time and Note Contents. When a note is selected on the Note List, its counterpart icon in the document window frame is highlighted.

An illustration of an example annotation is shown in Figure 2. The annotation, generally referenced 30, comprises a window having a title portion 32 and text portion 34. The window contains a note label (Sam), Note Owner ID or login name (Acme) and the data and time of the most recent synchronization.

5 A block diagram illustrating the collaborative document annotation system of the present invention is shown in Figure 3. The annotation system, generally referenced 40, comprises one or more standard web browsers 42 labeled web browser #1 through web browser #N. Each web browser comprises a Note Plug-In component that loads into the browser. The web browser 42 may comprise any suitable browser such as Netscape
10 Navigator or Microsoft Internet Explorer. Each web browser in combination with the Note Plug-In comprises a Notes Client and is the tool used for browsing the documents located on the server side. The Note Plug-In is invoked each time the browser gets a response from the Web Server 54 with a particular Multipurpose Internet Mail Extension (MIME) identifier. The communication protocol between Notes Clients and Notes Server is
15 standard Hyper Text Transport Protocol (HTTP). The web browsers are adapted to communicate over a global TCP/IP network such as the Internet 46. The web browsers 42 and the Notes Server 58 communicate with the web server 54 via the HTTP protocol.

The server portion of the systems 40 comprises a web server 54 with integrated Note Agent 56, Notes Server 58 coupled to a Notes Database 60, a Document File Server
20 62 and a messaging system 52, a Notes Administrator 50 and a Notes Viewer 48. The Notes Server communicates with the Notes Database via any suitable database language/protocol such as SQL. The Notes Server communicates with the messaging system via standard SMTP.

Annotation (Note)

25 An annotation or note is a portion of text or a graphical drawing that is associated with a specific location in a document. The terms annotation and note are intended to mean the same thing and are used interchangeably throughout this document. The location associated with a note in the document is called a Note Anchor and is kept separate from the annotation data itself. Once a note is created, its anchor point can be changed by the
30 user. The note anchor is expressed in terms of (X, Y) coordinates in the annotated page of a document. A note anchor may be set to a coordinate value of (0, 0) if the document associated with the note is a logical folder. Note that the annotation can be displayed

within its own window or can be layered on top of the displayed document. The first option is used in the case of text only annotations. The second option is used for mixed text and graphical annotations.

Notes Client

5 The application that displays the annotated document and allows for annotation sessions is called the Note Client and comprises the web browser 42 and Note Plug-In 44. In order to differentiate between notes generated by various users, each user chooses a unique user ID that forms a Note Owner Identifier. The Note Owner ID is formed from a combination of the chosen user name and user login name used when logging on to the
10 annotation system. The unique Note Owner ID is displayed in the note window title bar 32 (Figure 2) or when the Note Client application shows note properties for a graphical note, i.e., a note not within a window.

Notes are associated with specific documents that are stored on the server side. A note can belong to one document or to a set of documents aggregated under the same
15 logical structure. Documents and document aggregations are identified by their Note Document ID. The Note Document ID is a unique Universal Resource Locator (URL) on the World Wide Web.

It is important to note that the client portion of the annotation system of the present invention is not limited to implementation via a web browser and associated plug-in. The
20 invention can be implemented using any suitable means including, but not limited to, stand alone clients that implement the same functionality without the use of a web browser and/or plug-in software component.

Notes Server

The Notes Server 58 functions to provide the central document and note
25 management features. Normally the Notes Server functions in conjunction with a web server 54. It may access documents that reside in its own file system or that reside on remote file servers 62 located within the local area network. In addition, the Notes Server 58 may access these file servers and file systems by means of network file system protocols (represented by arrow 70). Once accessed, their contents are transmitted to the Notes
30 Client applications via TCP/IP protocols over communications means such as a LAN, WAN or the Internet 46. Note that the invention may be implemented to operate over an Intranet, e.g., LAN, or Extranet, e.g., WAN or the Internet.

An Annotation Session starts when a user logs onto the Web Server 54 by supplying a user name (user ID) and password. The Notes Server 58 requests this information when the user browses to a protected URL, which is the URL of a document to be viewed and/or annotated. The Notes Server 58 associates the user name and password
5 with the IP address of the request for identification purposes later in the Annotation Session. A session terminates when no activity is sensed by the Notes Server 58 for more than a predetermined period of time. Note that no special logout procedure is required to end an Annotation Session. The document protection mechanism can be set in combination with the web server security and the document file system security.

10 The Notes Server assigns a unique Note Serial Number upon the creation of each note. This number serves to uniquely identify each note in the system. A Note Event is defined as a message related to a note that did not exist at the time of the last synchronization with the Notes Server or a note that was modified or deleted since that time. Note that, at the client side, the user is permitted to create new notes, modify the
15 contents of notes, modify the anchor points of notes and delete notes. Only users with sufficient privileges, however, can modify notes. The system can be configured such that users without sufficient privileges are not permitted to view specific notes or to change them.

It is important to note that the server portion of the annotation system of the present
20 invention is not limited to implementation using a web server. The invention can be implemented using any suitable means including, but not limited to, a stand-alone HTTP server that implements the same functionality without the use of a web server. In addition, the invention is not limited to any specific client/server protocol implementation, e.g., HTTP.

25 Notes Synchronization

Note Events are exchanged between the server and the clients. Notes Clients forward to the Notes Server requests to view, modify, create or delete notes. In response, the Notes Server updates the clients with an up to date note list per document. This updating activity is termed Note Synchronization. The synchronization process is
30 performed asynchronously and is always initiated by the Notes Client in response to the user, i.e., it is user driven.

Alternatively, the synchronization process can be initiated by the Notes Client in a periodic manner, i.e., polling, or by the Notes Server when the exchange protocol in use allows for it. In the former case, the client would continuously poll the server. Utilizing Java in the server, rather than HTTP, permits the server to trigger the client and update the client display, if permitted to, since it is not practical for the server to poll clients. The server pushes new notes to selected clients by triggering them to pull new notes from the server. This serves to simulate a user pressing a synchronization button. This service can be provided for those clients that request it. Note that although the description contained herein refers the user driven synchronization process, the scope of the invention is intended to include other processes as well, such as server polling and server initiated via protocol means.

If the user elects to exit the display and annotation application or chooses to display and annotate another document without synchronizing the previous one, the Notes Client application saves the Note Events to the local file system. The next time the user browses to the same URL, the Notes Client application these events can be synchronized. Thus, the system allows for Note Persistence on the client side.

The Notes Server can be implemented as a process running on the web server computer. Alternatively, it is a process running on a separate computer that communicates with the web server. In either case, its functionality is the same. It is responsible for processing note events and maintaining the Notes Database and the Notes Log. The protocol the Notes Server utilizes is based on the well-known HTTP protocol. The Notes Server generates two independent HTTP streams: the first stream is for the displayed image and the second stream is for the annotations. Note again that alternatively, the invention can use any TCP/IP based protocol such as Java rather than HTTP.

The time of modification (in local client time) is stored as part of the Note Event. The Notes Server ignores the locally generated time stamp and records the time (in server time) of arrival for the event in the Notes Database and the Notes Log. The time stamp is termed the Note Time Stamp. The Notes Server calculates the difference between its own clock and the client clock to obtain the time zone of the client each time a Notes Synchronization is initiated at the client side. The time the Notes Client displays to the user for each event is the Note Time Stamp that was saved in the Notes Server adjusted to the local time zone of the client.

The administrator or other users (clients) of the Notes Server may elect to be notified each time a Note Event is received by the Notes Server. The notifications can be any suitable means of notification including, but not limited to visual alarms, audible alarms and/or automatic e-mail messaging sent to the administrator or the user of the system. These notification alerts are termed Note Alarms and are useful for document management workflow. Alarm notifications can be generated in accordance with various criteria including the type of Note Event, type of document. In addition, filtering maybe performed on a user basis or on a document basis.

Notes Server and Notes Database

On the Web Server 54, special address aliasing is provided by means of the Note Agent 56. The Note Agent 56 may be implemented as a Web Server Internet Server Application Programming Interface (ISAPI) filter or as a Web Server extension, both techniques being well known in the computer networking arts. All requests to aliased URLs are processed by the Note Agent 56 that in turn transfers the requests to the Notes Server 58.

The Notes Server 58 may reside in the same computer as the Web Server 54 computer or may reside in another computer located on the Local Area Network (LAN). The Notes Server 58 functions to receive modified URLs from the Note Agent 56 and provide in return individual documents and their associated annotations. If a document holds more than one page, the server may provide one page at a time to the client. Note that the documents may reside in the same file system as the Notes Server or in another file server 62 accessed over the local area network. The Notes Server 58 accesses the Document File Server 62 by means of standard Network File System (NFS) protocols.

The Notes Server 58 functions to keep track of all annotation activity in the Notes Database 60. The Notes Database 60 can be implemented in numerous ways. A convenient implementation is to select a standard SQL relational database. A Notes Table (not shown) in the Notes Database 60 is used to reflect the current state of the annotated documents. Transaction history is handled by a Notes Log Table (not shown) also stored in the Notes Database 60 which is implemented as a transaction log. All creation, modification and deletion events are logged to the Notes Log Table. The database is a useful tool for tracking all annotation and other activities on a per user or a per document basis. In addition, a viewer application such as Note Viewer 48 can optionally be used to

generate reports on the history of annotations on a per user or a per document basis that occurred over a period of time.

Each record in the Notes Table of the Notes Database 60 utilizes for the key index the unique Note Serial Number. Each record in the database comprises the following
5 information: Note Document ID, Note Contents (text or graphical information), Note Anchor coordinates, Note Time Stamp and Note Owner ID. This information is extracted from the Note Event data structure, which is described in more detail hereinbelow.

To simplify the implementation of the Notes Database 60, the Note Contents can be limited to a specific size. For example, the text of a note may be limited to a specific
10 number of bytes or characters, or the graphic drawing can be limited to a specific number of control points. Creating limits on the size of the note allows the record size to be predicted in advance. In some systems this may be important. The size limitation can be checked by the Notes Client or by the Notes Server when processing the Note Event transaction.

15 The Notes Server may be configured to utilize external messaging systems 52, such as an electronic mail server, for generating alert notifications to specific users. The users receiving the alert notifications may or may not be clients of the annotation system. The alert notifications may comprise any portion of the notes themselves along with additional information about the annotated document and the severity and/or urgency of the message.

20 The management of the Notes Server 58 and the Note Agent 56 in the Web Server 54 may be performed by a Notes Administration tool 50. One of the main functions of the Notes Administration tool is to provide the URL aliases needed for controlling access to specific documents.

Security and URL Aliases

25 Access permissions to the Notes Server 58 are defined on a per client user basis. As part of the task of administering the system 40, a directory is created for the user in the annotation root folder under the root directory of the web server 54.

The Note Agent 56 maintains two translation tables: a Document Location Table (Table 1) and a Client Access Key Table (Table 2). Both tables are present below.

Table 1: Document Location Table

| Alias | Actual Path |
|---------|-------------|
| Archive | D:\Arc |

Table 2: Client Access Key Table

| Client Name | Access Key | Location Alias | Sub Path |
|-------------|------------|----------------|------------|
| Acme | Acme_Docs | Archive | Acme_files |

5

In order to access the system, a client needs to provide the Note Agent Trigger Token as part of the URL. In the example presented herein, this is the token 'annotate.' This Trigger Token is used by the Web Server 54, e.g., Microsoft Internet information Server (IIS), to invoke the filter in the Note Agent 56 to translate the document path specified in the remainder of the URL. The requested URL is as follows below.

10 Requested URL = http://domain_name/annotate/Acme_Docs/Subfolder/document.name
where

domain name = 'domain_name'

Note Agent Trigger Token = 'annotate'

15 Customer Access key = 'Acme_Docs'

In accordance with the above described aliasing mechanism, the system 40 may ask users to type in their unique user name and password in order to obtain permission to browse particular locations or file systems in the local area network on the server side. In this manner, users can access their private documents for display and annotation purposes while not violating the privacy and security of other users of the system.

20 For example, assuming a Microsoft IIS Web Server 54 that defines 'C:\inetpub\wwwroot' as the base document root, the Notes Server 58 will utilize a sub folder called for example, 'annotate.' This specific name 'annotate' would trigger the Note Agent ISAPI filter 56 in the Web Server 54 to translate the URL as specified below.

25 If 'Acme' is a name of a user of the system, a folder named 'Acme_Docs' will be created under the 'C:\inetpub\wwwroot' directory. This folder is the Client Protected

Folder. Using the operating system security flags, only a user named 'Acme' would be given rights to browse this folder.

Client Protected Folder = 'C:\inetpub\wwwroot\annotate\Acme'

5 The Note Agent 56 functions to provide mechanisms for translating URL addresses under the 'Acme_Docs' folder in such a way that logical names can point to physical folders on the same computer or on other computers located in the network. In connection with the example presented above, the Client Document Root physical path is as follows.

Client Document Root physical path = 'd:\Arc\Acme_files'

The Client Document Root physical path is translated to the following URL.

10 Translated URL = 'http://domain_name/annotate/Acme_Docs'

As result, the 'Acme' client is granted permission to access any document under her/his Document Root folder. Note that neither the Document Location Table (Table 1) nor the Document Access Key Table (Table 2) is limited in length. Thus, a client may have more than one access key. In addition, there may be more than one alias associated with
15 the same physical folder; each alias providing protection for a different client.

Annotation Session: Initialization

The following describes the initialization portion of an Annotation Session in more detail. A flow diagram illustrating the initialization portion of a general Annotation Session is shown in Figure 4. Note that the client/server protocol used throughout the
20 session is the HTTP protocol.

The Session starts when the user launches a web browser and provides a URL in the form that was described above and in Figure 3. This URL is referring to a directory or to a specific document. The document may contain one page or more. An example for a single page document is a TIFF file format image. An example of a multi-page document is
25 Adobe Portable Document Format (PDF).

Initially, the user launches her/his web browser (step 80). To initiate the initial request, the client, via her/his web browser provides a URL (step 82) which is transmitted to the server (step 84). The server verifies whether the client was already authenticated (step 86). If the client was not previously authenticated, the server requests authentication
30 from the user in response (step 88). To authenticate a user, the browser first displays a dialog to the user, asking for a user name and password. The browser then transmits the user-supplied information to the server (step 90). The web server then checks the

credentials submitted against those defined for the requested URL (step 92). The construction of the URL and associated aliases was described previously in the section titled Security and URL Aliases.

5 If the user name and password provided by the user are not correct (step 94), an error message is displayed (step 99) and the authentication portion is repeated (step 88). The user is then queried again for the correct user name and password. If the user name and password provided by the user are correct (step 94), the server responds with the directory listing of the URL requested by the user or with the contents of a document if the URL was referring to a specific document (step 96).

10 The client then displays the directory listing in a window within the web browser (step 98). It is likely that the directory contains documents that the user may wish to annotate at some point in time. Note that the list of documents sent by the server when browsing is first filtered by the Notes Server 58 (Figure 3) to display only those documents with file types that the Note Server knows how to interpret.

15 A message flow diagram illustrating the transaction sequence between client and server during initialization is shown in Figure 5. As described above, the first request (referenced 100) is made by the client with the requested URL to be displayed. The server first verifies the authentication of the user. If no user name or password was submitted, the server responds with an authentication request (referenced 102). Once received by the
20 client, the user is queried for a user name and password. The user-supplied input is sent as another request to the server (referenced 104). The server attempts to verify the authentication and if it passes, the server first filters and then sends the directory listing associated with the requested URL to the client (referenced 106). Upon receiving the directory listing, the client displays its contents in window in the web browser.

25 Annotation Session: Browsing

The following describes the browsing portion of an Annotation Session in more detail. A flow diagram illustrating the document browsing portion of a general Annotation Session is shown in Figure 6. Note that the client/server protocol used throughout the session is the HTTP protocol.

30 The client first issues a request to the server to view a particular document by selecting its corresponding document URL (step 110). Before the server responds with the requested data, it needs to validate the user's credentials once again since this request to

view a document is considered a new connection to the server (step 112). Assuming the user's name and password were supplied previously, the web browser provides the information automatically, thus avoiding an error message response from the server. If the user is not authenticated (step 114), an error message is returned and displayed on the client
5 browser (step 116). The document is not transmitted to the user until the user is authenticated. The mechanics of user name and password authentication by the server was described in detail previously in the Initialization section.

Once authenticated, the server then responds by sending the document data, i.e., image data in this example, to the client with the proper MIME type (step 118). The
10 MIME type sent to the client in response to the request is unique and is the same for all document types. In response to receiving the MIME type, the web browser at the client, loads the Note Plug-In 44 (Figure 3) on the client (step 120). The Note Plug-In client functions to take control of the browser display window and begin displaying the image (step 122).

15 Immediately thereafter, the client issues a request to retrieve the notes associated with the image, if there are any, from the server (step 124). The URL that is supplied to the server to retrieve the notes is the same URL corresponding to the underlying document (image) but having a suffix of '\notes' appended to it. Upon receiving the appended URL, the server searches the Notes Database 60 (Figure 3) for the specified document using the
20 URL as the key for searching (step 126). Once found, the server filters the notes so as to supply only those notes that the requesting client has permission to view (step 128). The server subsequently serializes the notes after filtering and stores them in a response buffer (step 130). The notes buffer is then sent as a response to the client (step 132). The response buffer data structure is described in more detail hereinbelow.

25 Upon receipt of the notes buffer, the client updates its locally stored Notes Database (step 134) and then displays the contents of the notes buffer in a window within the web browser (step 136). If the user has selected to view the annotations overlying the document, i.e., the image, then the client displays the notes similarly to that shown in Figure 1B. The user then can process one or more notes, i.e., create new notes, edit
30 existing ones and/or delete one or more notes (step 138).

The client is now ready to display notes and accept modification to them. The Notes Server has supplied the number of notes in the Server Annotation Response Data Response Structure (Figure 12). If the number of notes is not equal to zero, the client

checks the state of the viewing condition flag in the web browser Note Plug-In application. This variable may have the following values: show document only (corresponding to Figure 1A), show document in addition to notes (corresponding to Figure 1B), show document and notes in one web browser frame and a list of notes in another web browser frame (corresponding to Figure 1C).

In case the third option is selected, another web browser frame is created and the document is redrawn in the right frame and the list of notes is drawn in the left frame. The notes displayed in the list in the left frame can be sorted and filtered using criteria such as, for example, time of creation, Note Owner ID, Note Contents, note shape type, and note color.

In accordance with the invention, the client is capable of displaying the notes as they arrive from the server regardless of the state of the document image stream display. This allows the user to begin manipulating the annotations when large image documents are still in the midst of being transmitting from the server.

A message flow diagram illustrating the transaction sequence between client and server during document browsing general Annotation Session is shown in Figure 7. As described previously, the client initiates a document browsing session by sending the document URL request to the server (referenced 140). The server then validates the user permissions including user name and password. Once the user is authenticated, the server responds to the client with the document MIME type and the corresponding document data (image data in this example) (referenced 142).

The client checks the MIME type and, in response, loads the Note Plug-In into the web browser. Once loaded, the image is displayed in a window in the browser. Then, the annotations associated with the image are requested from the server by sending the URL corresponding to the document with the text '/notes' appended to it (referenced 144). The server searches the Notes Database and filters the notes for those corresponding to the image that the user has permissions for. The resulting groups of notes are serialized and placed in a notes buffer. The buffer is then transmitted to the client (referenced 146). The client receives the buffer, updates the local note database, displays the notes in a window in the browser and gets updates to one or more notes from the user.

In accordance with the invention, the Note Plug-In running on the client web browser provides the user with the capability to perform several operations on notes. In particular the client permits a user to perform the following.

Notes can be created by selecting an annotation tool on the screen and placing the cursor in the area of the image where it is desired to place a note. A box is created and the user can enter text. Once entered, the text appears in the form of a 'sticky' note that appears on top of the image. The Note Anchor is created at the location the user placed the note. Notes can be viewed by double clicking, for example, on a note icon. Note icons are displayed on the screen when the viewing option is set to display both notes and the image. A note can be closed after viewing by clicking on the box in the upper left-hand corner of the title bar 32 (Figure 2). Note that alternatively, graphics can be entered into a note rather than text. In this case, a graphics tool or marker is used to place graphics in the note.

A note can be moved to a new location by grabbing a note icon and dragging it to a new location on the image. The Note Anchor is changed to correspond to the new location. Other features include printing the document with the notes attached; scrolling to another section of the document with the notes scrolling with the document; zooming in or out within the displayed document; deleting a note; resizing a note that contains either text or a graphic drawing; changing the default note color, dimensions and associated default marker. Further, notes can be synchronized with the Notes Server, which is described in more detail hereinbelow.

It is important to note that a user may create as many notes as she/he desires. Existing notes can be modified even if other users created them. The server, however, will not permit any changes that the user is not authorized to make. In accordance with the invention, all changes made to notes are saved in a local note database on the client. The database is managed within the RAM of the computer and is backed up on the local computer mass storage device, i.e., hard disk drive.

This backup of the database also functions to provide a recovery mechanism in case the user exits the application or requests a different document URL without synchronizing with the server beforehand. The database is kept for a limited period of time before being erased. The time period is set by the user and is typically on the order of a few days. While the database is saved, the user retains the option of synchronizing any unsaved notes the next time she/he browses to the same document URL.

The structure of the local note client database is similar to that of the server note database with the exception that no authentication checks are performed on the client. Thus, for example, the note database on the client does not contain a Note Owner password field.

Annotation Session: Synchronization

The following describes the synchronization portion of an Annotation Session in more detail. A flow diagram illustrating the note synchronization portion of general Annotation Session is shown in Figure 8. The process of synchronizing notes is initiated and driven by the user. The user initiates the synchronization process, for example, by pressing a button in the browser application called 'Sync' (step 150). The Notes Client, in response to the button press, functions to prepare a note buffer to send to the Notes Server (step 152). The note buffer contains only the notes that were changed since the time of the previous synchronization event. The time of the previous synchronization event is taken as either the time of the initial request to retrieve the current document or the actual time the user previously pressed the 'sync' button in the Notes Client application. The time of the last synchronization event is always known and stored by the Notes Client application. This time is also transmitted by the Notes Server in the Server Annotation Response Data Structure.

The client then serializes the notes within the note buffer (step 154). After serialization of the note data, the client uses the data in the buffer to construct a Client Annotation Event Data Structure (Figure 10) and POSTs it to the Notes Server using the original URL provided by the user to retrieve the notes previously (step 156). The server validates the user name and password (step 158) and merges the note event information with the Notes Database 60 (Figure 3) (step 160). The merge process includes checking whether or not the user has the required permission level to modify notes. Those notes, i.e., Note Events, that the user is not authorized to perform are ignored. Any new notes to be created, as indicated by a zero in the Note Serial Number field, are assigned a unique Note Serial Number. A new record is created in the Notes Database and the new note is placed in the record (step 162). A new record is also created in the Notes Table. The transaction is also recorded in the Notes Log Table (step 164).

Next, the Notes Server prepares a Server Annotation Response Data Structure filled with the new list of notes and their content (step 166). An important aspect of the invention is that the list of notes includes changes made by all concurrently acting clients that have previously synchronized their notes with the Notes Server. In this fashion, the annotations associated with a particular document made by all other clients are replicated to each client at the time they perform note synchronization.

The Notes Server responds to the synchronizing client by transmitting the Server Annotation Response Data Structure to the client (step 168). Once received, the client merges the notes list with its local notes database (step 170). The display is then refreshed and the Notes Client is ready to receive and process input from the user once again (step 172).

A message flow diagram illustrating the transaction sequence between client and server during note synchronization is shown in Figure 9. As described previously, the synchronization process is initiated and driven by the user. Once initiated by the user, the client prepares a notes buffer with the notes to be processed by the Notes Server and POSTs, i.e., transmits, the buffer to the server (referenced 180). The notes are sent in the form of a Client Annotation Event Data Structure. The Notes Server, in response, updates its Notes Database after authenticating the user. Any actions requested by the user that are not authorized due to lack of permissions, are ignored.

The actions are processed by the server and a response buffer is created. The buffer is sent to the client as a Server Annotation Response Data Structure (referenced 182). The client, upon receipt of the notes list, updates its local notes database, displays the notes in a frame in the web browser and inputs one or more notes commands from the user. The process is repeated while the user continues to create, edit and delete notes.

Client Annotation Event Data Structure

A diagram illustrating the Client Annotation Event Data Structure in more detail is shown in Figure 10. The data structure, generally referenced 190, comprises a plurality of fields for conveying note data from the Notes Client to the Notes Server. The data structure 190 comprises a Document ID field 192 which comprises the URL of the annotated document the notes are associated with, local time 194 of the client along with the time zone of the client, the Note Owner ID 196 which uniquely identifies the user and the number of notes 198 contained in the message. Following the number of notes is a buffer containing one or more notes labeled note #1 200 through note #N 202.

A diagram illustrating the note data structure of the Client Annotation Event Data Structure of Figure 10 in more detail is shown in Figure 11. The data structure, generally referenced 210, comprises a plurality of fields that are used by the client to represent each note. Each note comprises a Note Serial Number 212 field uniquely identifying the note. The Note Serial number is set to zero for a new note and set to a non-zero value for an

existing note. An action field 214 represents the action to be taken on the note. The following values are valid actions.

- | | | |
|---|---|---------------|
| | 0 | No action |
| | 1 | Add a note |
| 5 | 2 | Modify a note |
| | 3 | Delete a note |
| | 4 | Add a comment |
| | 5 | Update status |

A Note Anchor field 216 stores the coordinates of the position of the note in the document.

- 10 The note status field 218 can have the following values.

- | | | |
|--|---|-----------------|
| | 0 | Ignore |
| | 1 | To be performed |
| | 2 | Done |

- 15 The note color field 220 represents the color the note is to be displayed in. The note color represents a RGB value.

The text length of the note 222 represents the number of bytes or characters of the text note itself. The note text field 224 represents a buffer with the actual text of the note.

The note shape field 226 can have the following values.

- | | | |
|----|---|-----------|
| | 0 | No shape |
| 20 | 1 | Polygon |
| | 2 | Spline |
| | 3 | Rectangle |
| | 4 | Ellipse |
| | 5 | Marker |

- 25 A note shape line cap field 228 can have the following values.

- | | | |
|----|---|-------------|
| | 0 | None |
| | 1 | Right arrow |
| | 2 | Left arrow |
| | 3 | Both arrows |
| 30 | 4 | Round |
| | 5 | Square |

The note shape line width field 230 represents the width of the shape line in millimeters.

The note marker type field 232 represents the type of marker and can have the following values.

- | | | |
|----|---|-----------------|
| 35 | 0 | No marker |
| | 1 | Star |
| | 2 | Circle |
| | 3 | 'Approved' sign |
| | 4 | 'Rejected' sign |
| 40 | 5 | 'Correct' sign |
| | 6 | 'Remove' sign |

The number of shape vertices field 234 represents the number of points in the shape and the shape vertices 236 comprise a plurality of N (x, y) coordinates.

Server Annotation Response Data Structure

A diagram illustrating the Server Annotation Response Data Structure in more detail is shown in Figure 12. The data structure, generally referenced 240, comprises a plurality of fields for conveying note data from the Notes Server to the Notes Client. The data structure 240 comprises a Document ID field 242 which comprises the URL of the annotated document the notes are associated with, local server time 244, the dimensions of the document 246 including the width and height in millimeters and the resolution of the width and height in pixels per millimeter and the number of notes 248 contained in the message. Following the number of notes is a buffer containing one or more notes labeled note #1 250 through note #N 252.

A diagram illustrating the note data structure of the Server Annotation Response Data Structure of Figure 12 in more detail is shown in Figure 13. The data structure, generally referenced 260, comprises a plurality of fields that are used by the server to represent each note. Each note comprises a Note Serial Number 262 field uniquely identifying the note. The Note Serial number is set to zero for a new note and set to a non-zero value for an existing note. The Note Time Stamp field 264 represents the time (in server time) of the update for this particular note. An action field 266 represents the action to be taken on the note. The following values are valid actions.

- | | | |
|----|---|---------------|
| | 0 | No action |
| | 1 | Add a note |
| | 2 | Modify a note |
| | 3 | Delete a note |
| 25 | 4 | Add a comment |
| | 5 | Update status |

A Note Anchor field 268 stores the coordinates of the position of the note in the document. The note status field 270 can have the following values.

- | | | |
|----|---|-----------------|
| | 0 | Ignore |
| 30 | 1 | To be performed |
| | 2 | Done |

The note color field 272 represents the color the note is to be displayed in. The note color represents a RGB value.

The text length of the note 274 represents the number of bytes or characters of the text note itself. The note text field 276 represents a buffer with the actual text of the note.

The note shape field 278 can have the following values.

- | | | |
|---|---|-----------|
| | 0 | No shape |
| 5 | 1 | Polygon |
| | 2 | Spline |
| | 3 | Rectangle |
| | 4 | Ellipse |
| | 5 | Marker |

10 A note shape line cap field 280 can have the following values.

- | | | |
|----|---|-------------|
| | 0 | None |
| | 1 | Right arrow |
| | 2 | Left arrow |
| | 3 | Both arrows |
| 15 | 4 | Round |
| | 5 | Square |

The note shape line width field 282 represents the width of the shape line in millimeters.

The note marker type field 284 represents the type of marker and can have the following values.

- | | | |
|----|---|-----------------|
| 20 | 0 | No marker |
| | 1 | Star |
| | 2 | Circle |
| | 3 | 'Approved' sign |
| | 4 | 'Rejected' sign |
| 25 | 5 | 'Correct' sign |
| | 6 | 'Remove' sign |

The number of shape vertices field 286 represents the number of points in the shape and the shape vertices 288 comprise a plurality of N (X, Y) coordinates.

While the invention has been described with respect to a limited number of
 30 embodiments, it will be appreciated that many variations, modifications and other applications of the invention may be made.

CLAIMS

1. A system for annotating documents, comprising:
 - a document file located on a server, said document file for storing one or more documents;
 - 5 a notes database located on said server, said notes database for storing one or more notes, each note associated with a particular document;
 - one or more notes clients coupled to a network, each notes client operative to locally display a representation of a document remotely stored on said server in said document file, said notes client adapted to permit a user to annotate
 - 10 said document with one or more notes, said notes client operative to simultaneously display said one or more notes associated with said document over the displayed document such that said document is viewable along with said one or more notes;
 - a notes server coupled to said network, said notes server operative to store said
 - 15 documents in said document file separately from notes stored in said notes database, said notes server receiving one or more notes associated with a particular document from said one or more notes clients; and
 - synchronization means within said notes clients and said notes server, said synchronization means for updating said notes server with any notes events
 - 20 processed by said notes clients and for updating said notes client with the results of synchronization updates previously performed by other notes clients since the last synchronization event.
2. The system according to claim 1, wherein said notes database comprises a Notes Table for storing the current content of a note.
- 25 3. The system according to claim 1, wherein said notes database comprises a Notes Log Table for storing a transaction history of all note events for notes associated with a particular document.
4. The system according to claim 1, said notes client comprises:
 - a display;
 - 30 network communication means;

a processor suitably programmed to run:

a web browser; and

a client note plug-in adapted to run within said web browser.

5. The system according to claim 1, wherein said network comprises the Internet.

5 6. The system according to claim 1, wherein said network comprises a TCP/IP based network.

7. The system according to claim 1, wherein said network comprises an Intranet.

8. The system according to claim 1, wherein said notes server comprises:
network communication means;

10 a processor suitably programmed to:

provide web server services;

provide notes agent services; and

provide document and notes browsing and synchronization services to said
one or more notes clients.

15 9. The system according to claim 8, wherein said notes agent services comprises an Internet Server Application Programming Interface (ISAPI).

10. The system according to claim 1, further comprising a notes administrator for providing one or more Universal Resource Location (URL) aliases in managing and administering said notes server.

20 11. The system according to claim 1, further comprising a note viewer for generating reports on the history of annotations on a per user or a per document basis that have occurred over a period of time.

12. The system according to claim 1, wherein two or more user can collaboratively work on the same document whereby each user is capable of viewing annotations made by
25 the other users upon synchronization with said notes server.

13. The system according to claim 1, wherein a single page of a multipage document can be annotated on a notes client without requiring the entire multipage document to be transmitted to said notes client.

14. The system according to claim 1, further comprising security means for providing Universal Resource Locator (URL) translation and redirection services so as to provide user level security for documents and notes stored on said server.
15. The system according to claim 14, wherein said security means comprises a Document Location Table and a Client Access Key Table.
16. The system according to claim 1, wherein said document comprises an image.
17. The system according to claim 1, wherein said document is represented using a page description language.
18. The system according to claim 1, wherein said notes client is adapted to permit said user to annotate said document in parallel with the display of said document.
19. The system according to claim 1, wherein said document comprises a plurality of pages whereby each page can be annotated independently of all the other pages.
20. The system according to claim 1, wherein said synchronization means is user initiated.
21. The system according to claim 1, wherein said synchronization means is triggered by said notes server via the signaling of selected notes clients.
22. A method of annotating documents, said method comprising the steps of:
getting a first request comprising a Universal Resource Locator (URL) associated with a document to be viewed and annotated and transmitting said first request from a client to a server;
sending a first response from said server in response to said first request, said first response comprising the document type of and a representation of said requested document;
getting a second request comprising a document note URL corresponding to the notes associated with said document and sending said second request from said client to said server;

5 sending a second response from said server in response to said second request, said
 second response comprising one or more notes associated with said
 document;
 displaying the representation of said document and the associated one or more notes
10 locally on said client such that both are viewable simultaneously;
 permitting said user to perform annotation functions on said client so as to generate
 a third request comprising one or more note events related to said document
 that are transmitted from said client to said server; and
 sending a third response from said server to said client in response to said third
15 request, said third response comprising an updated list of notes associated
 with said document, said list of notes incorporating any note events
 previously received from other clients.

23. The method according to claim 22, wherein said document type comprises a
Multipurpose Internet Mail Extension (MIME) corresponding to said document.

15 24. The method according to claim 22, wherein said step of getting a first request
comprises the step of querying said user for a user name and password.

25. The method according to claim 22, wherein said step of sending a first response
comprises the step of authenticating said user on said server before sending said first
response.

20 26. The method according to claim 22, further comprising the step of getting a fourth
request comprising a folder URL associated with the folder comprising one or more
documents to be viewed and annotated and transmitting said fourth request from said client
to said server.

27. The method according to claim 22, further comprising the step of sending a fourth
25 response from said server in response to receiving a request comprising a folder URL, said
 fourth response comprising a directory listing of said folder.

28. The method according to claim 22, wherein said document comprises an image.

29. The method according to claim 22, wherein said document is represented using a
page description language.

30. The method according to claim 22, wherein said step of permitting said user to perform annotation functions on said client notes client occurs simultaneously with the display of said document.

5 31. The method according to claim 22, wherein said document comprises a plurality of pages whereby each page is capable of being annotated independently of all the other pages.

32. The method according to claim 22, wherein said steps of generating a third request and receiving a third response is initiated by said user.

10 33. The system according to claim 22, wherein said steps of generating said third request and receiving said third response is initiated by said server via the signaling of each selected client.

1/13

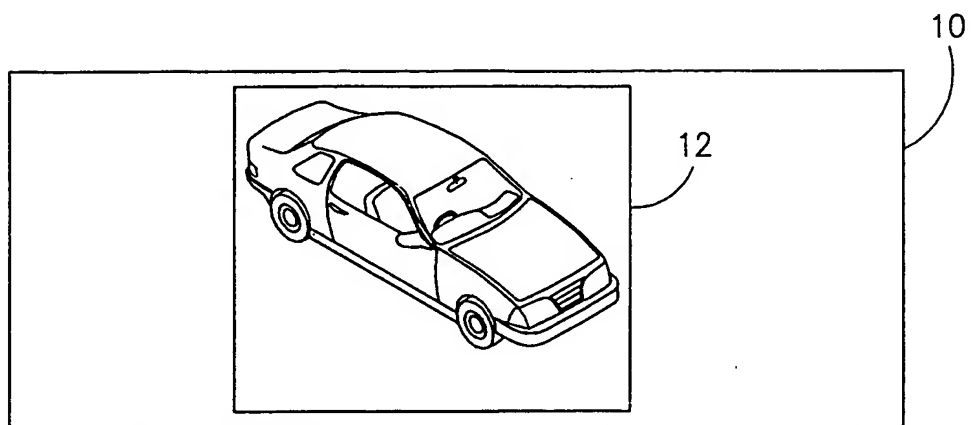


FIG. 1A

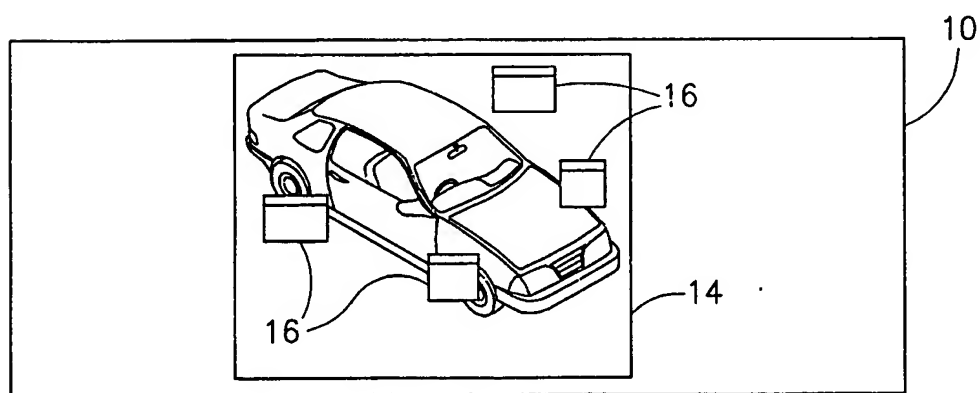


FIG. 1B

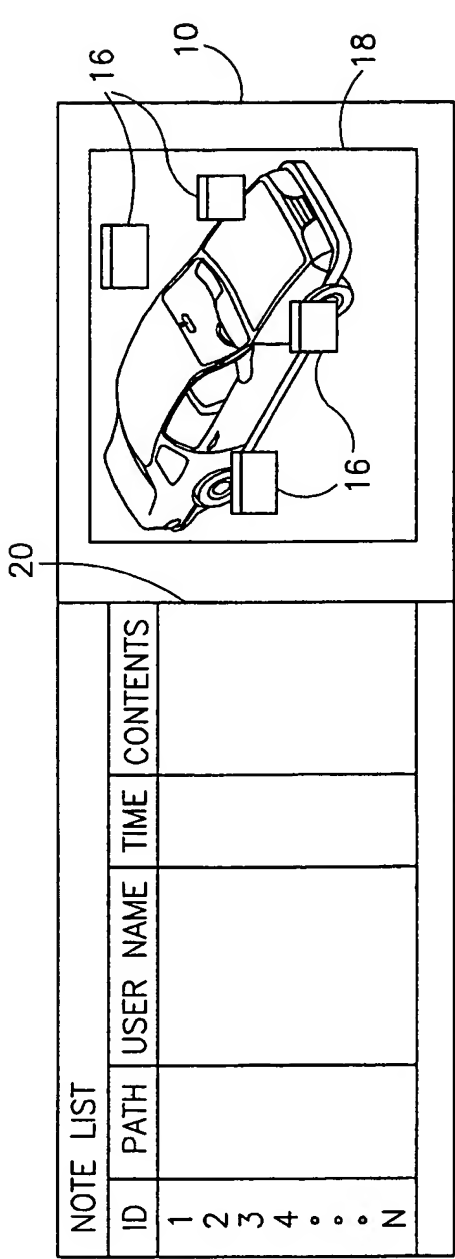


FIG.1C

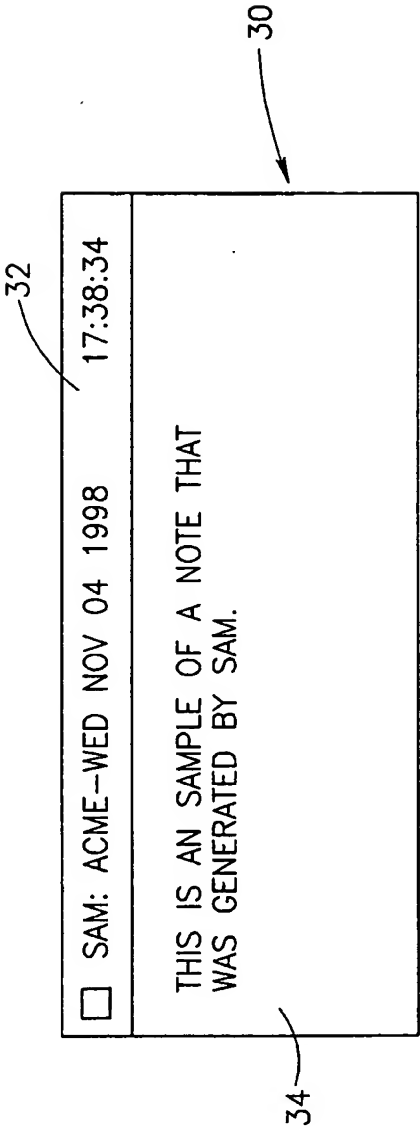
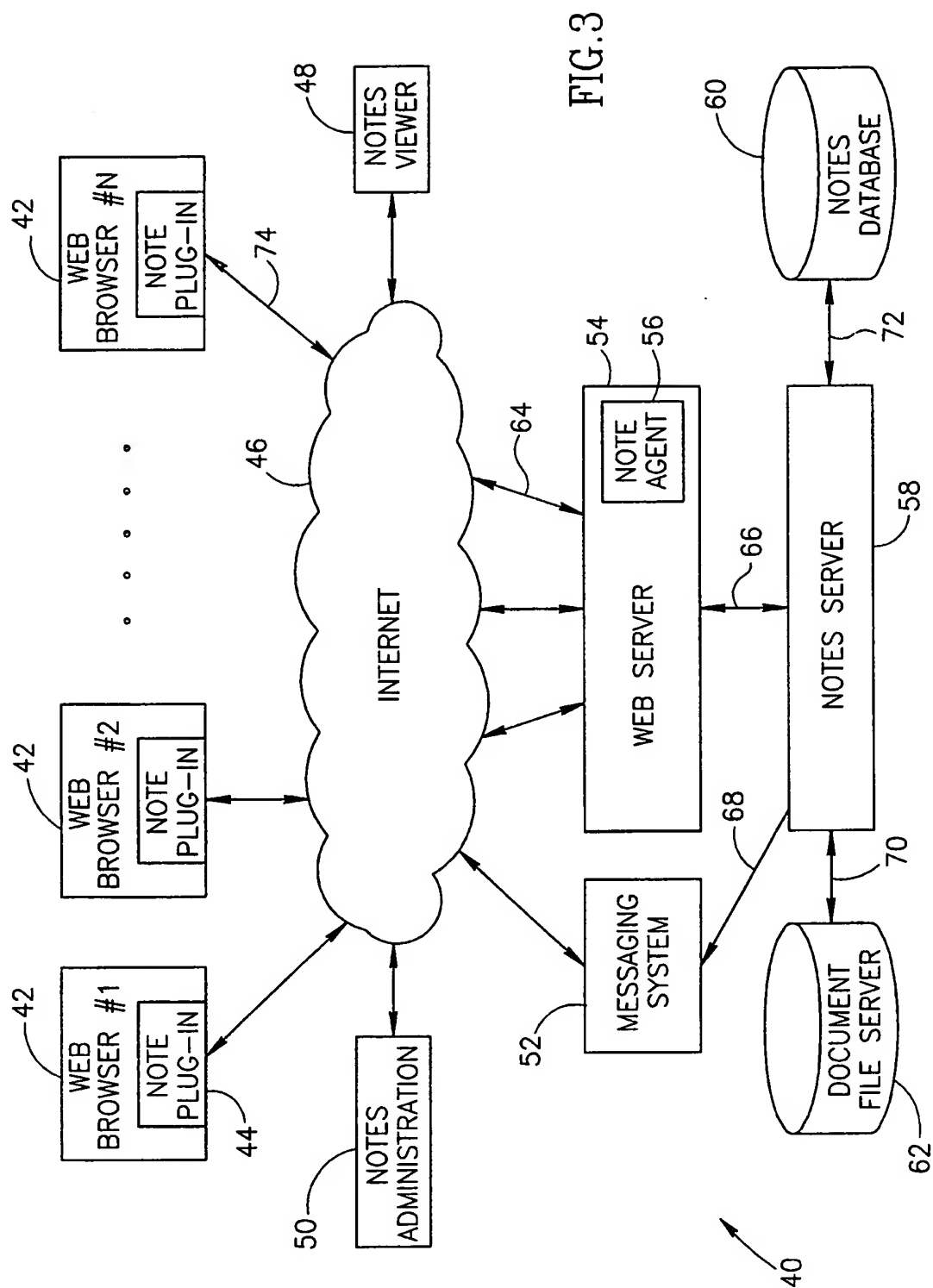


FIG.2

3/13



4/13

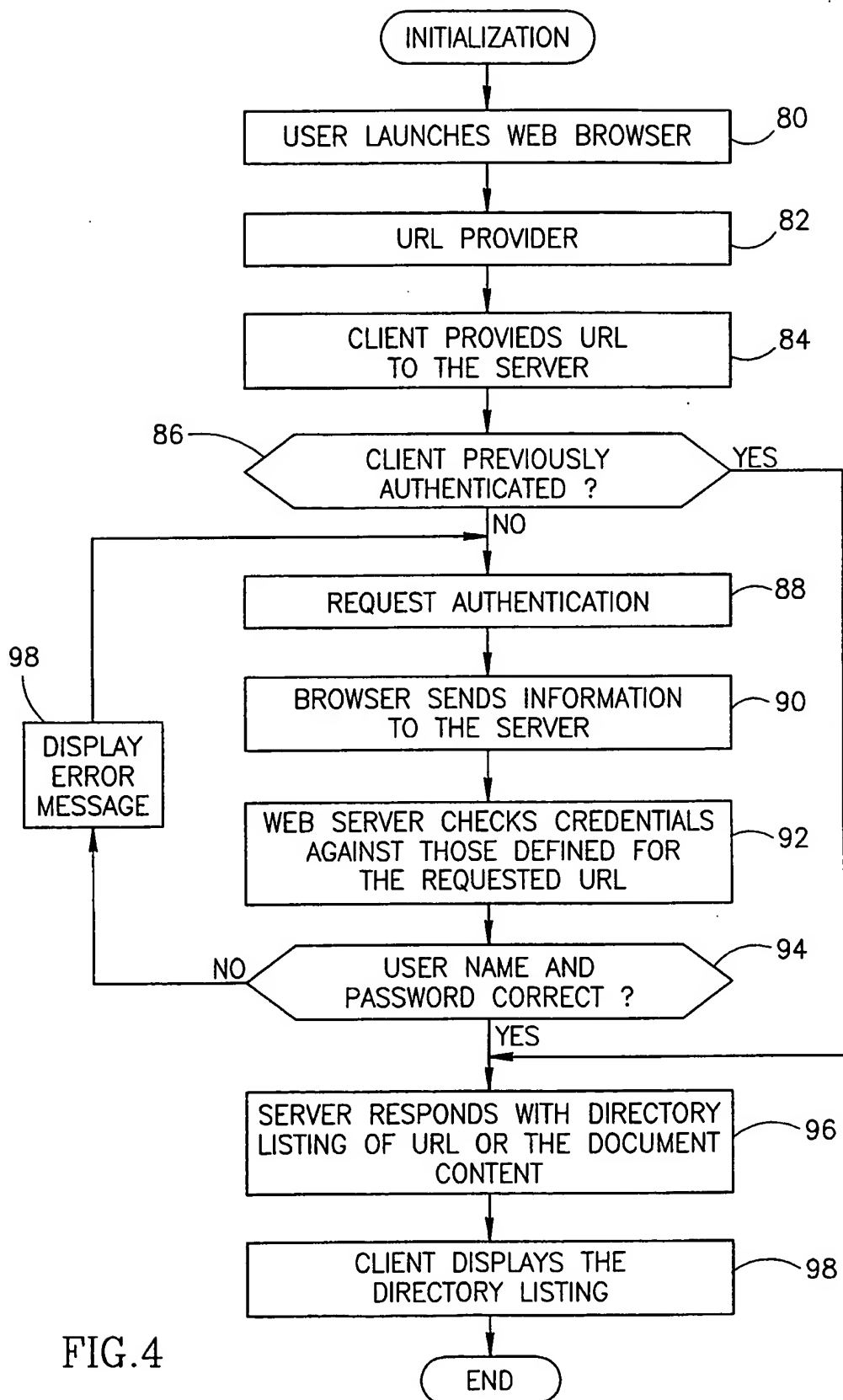


FIG.4

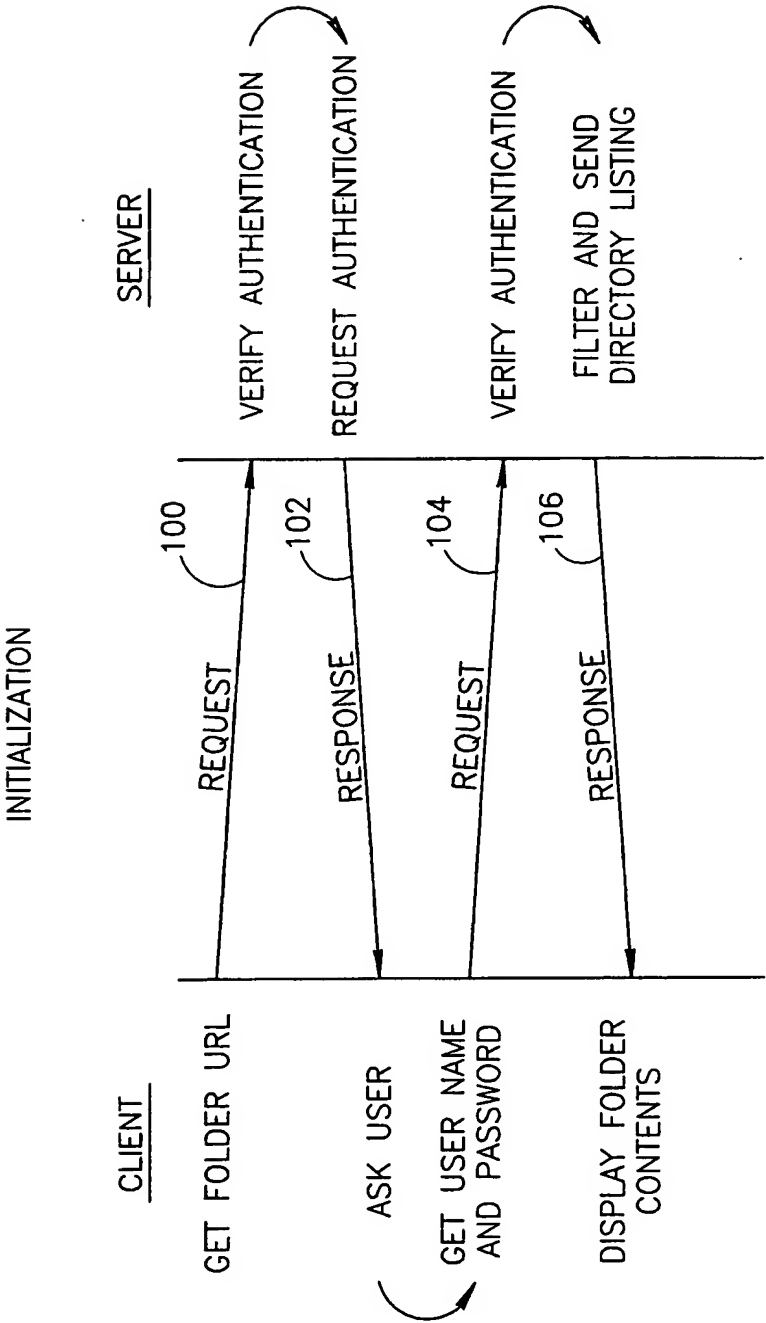


FIG.5

6/13

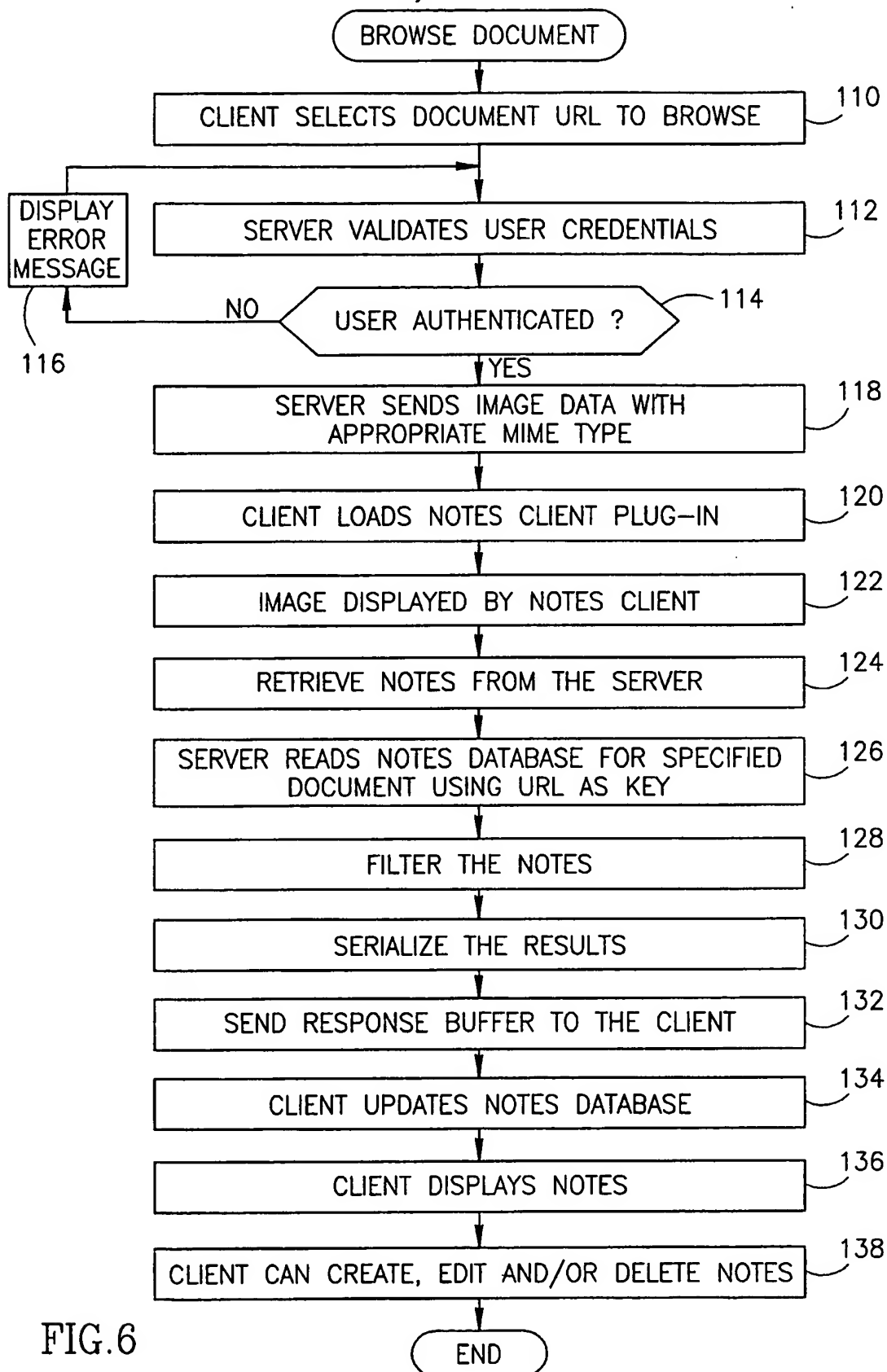


FIG. 6

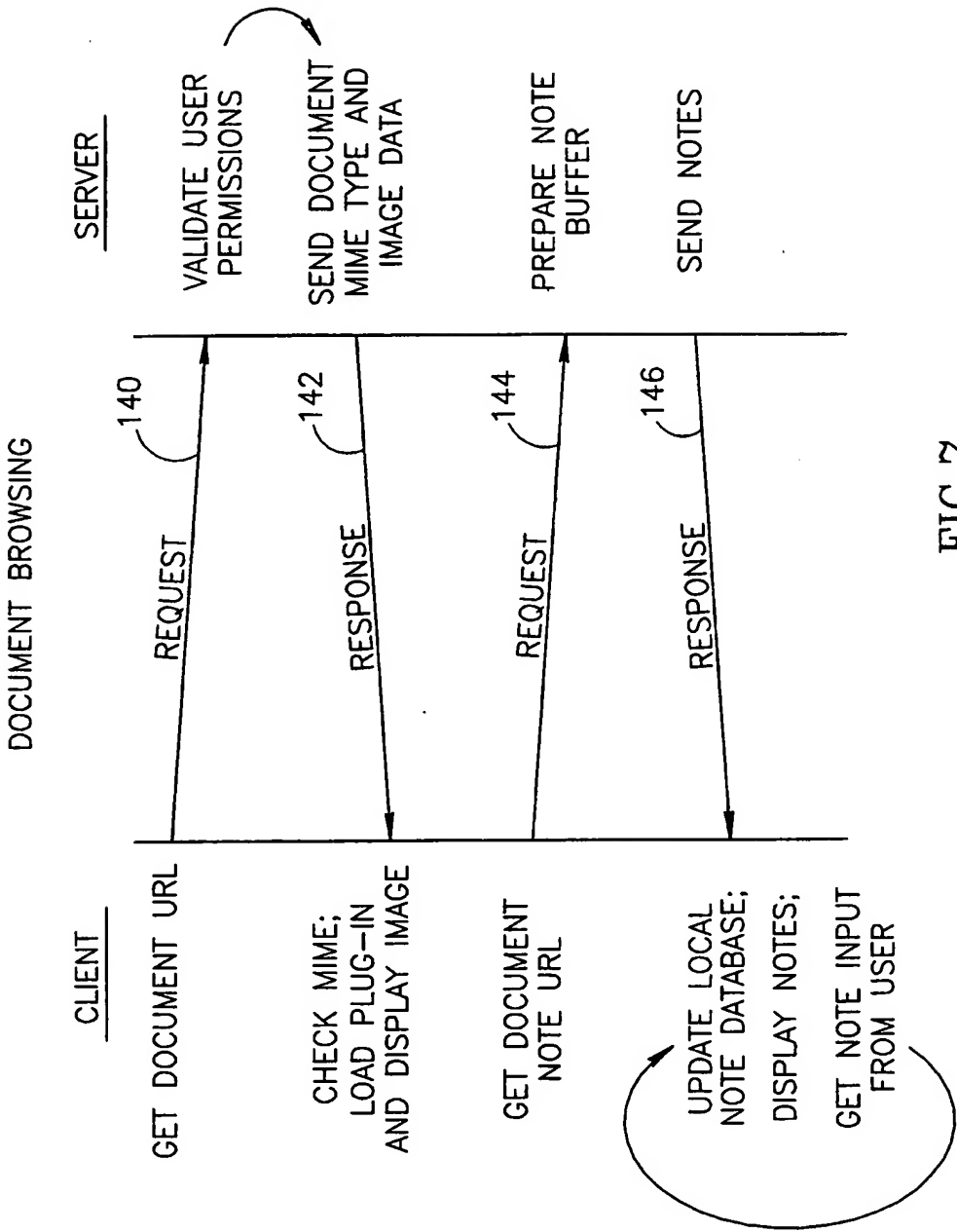


FIG. 7

8/13

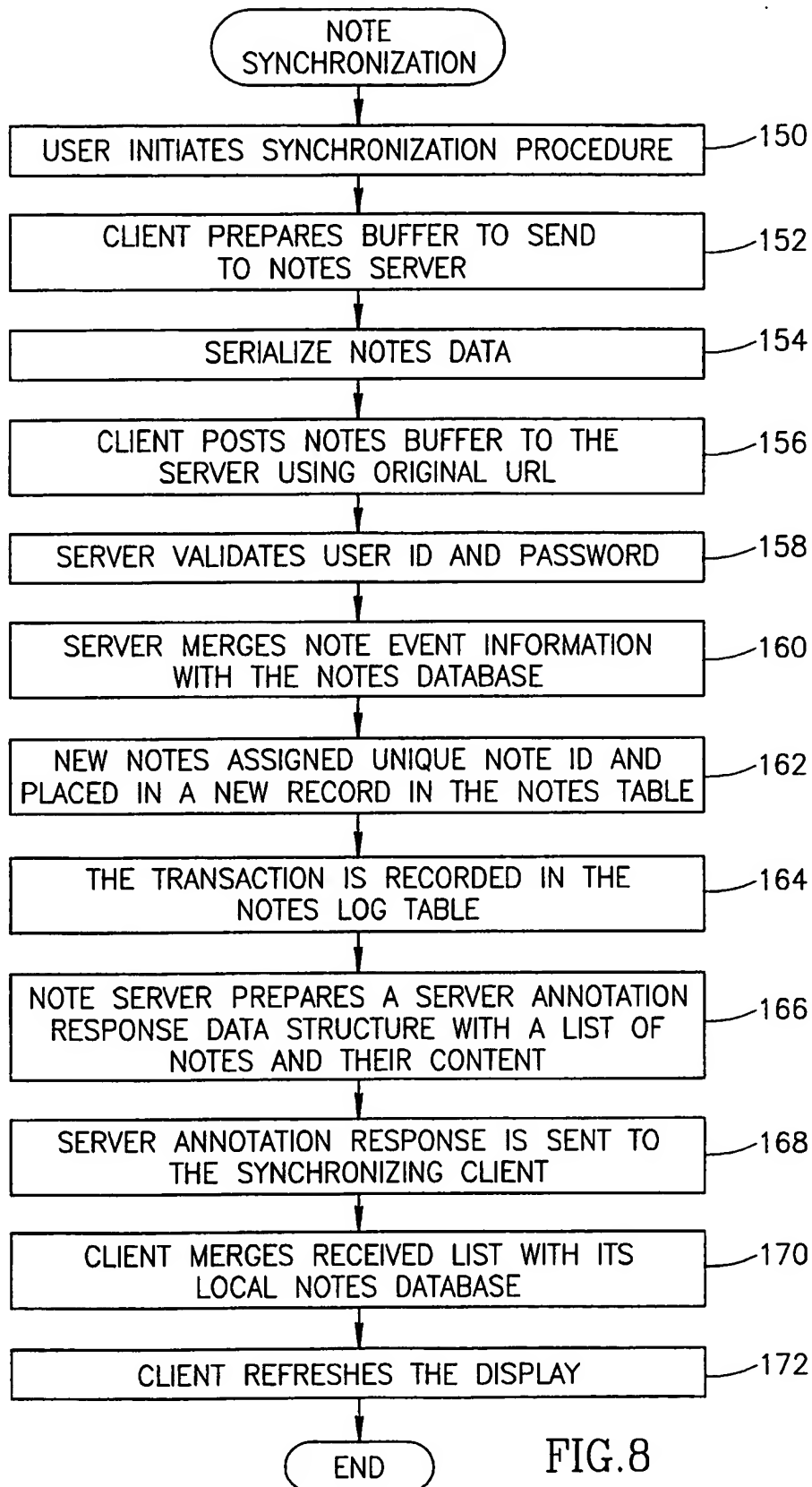


FIG.8

9/13

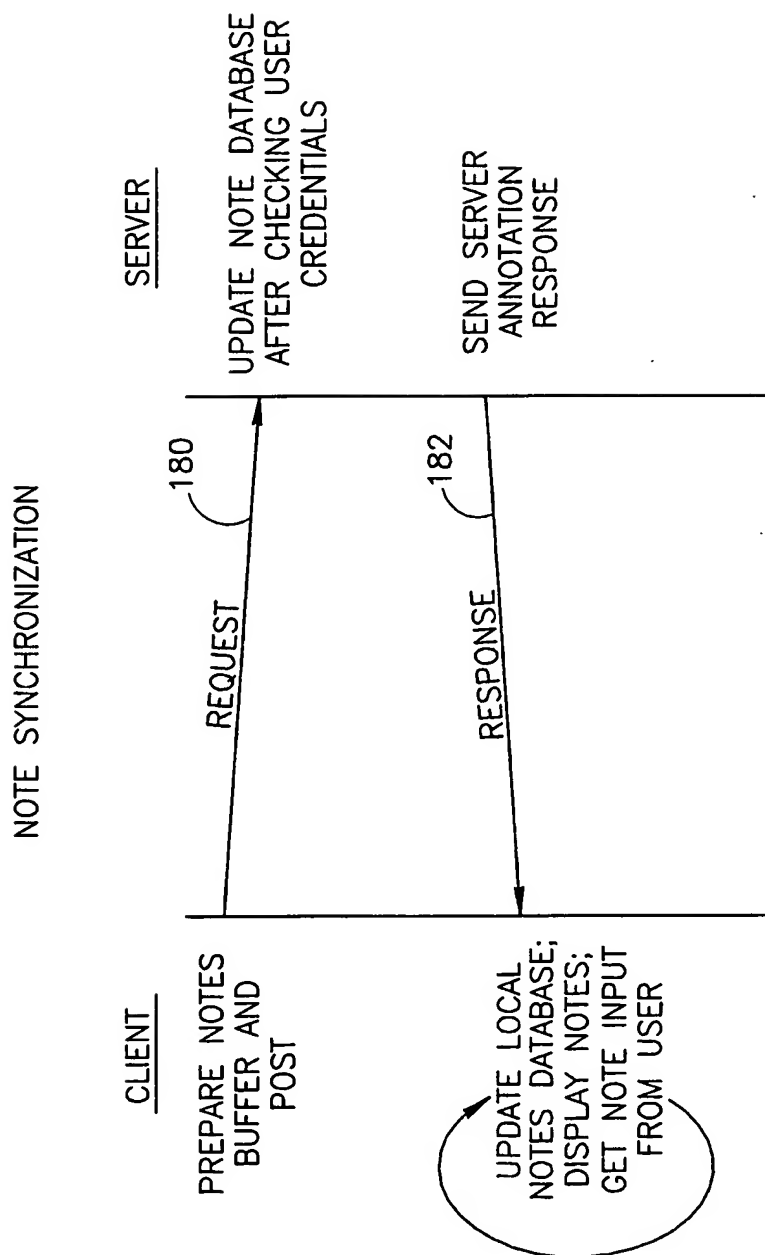


FIG.9

10/13

CLIENT ANNOTATION EVENT DATA STRUCTURE

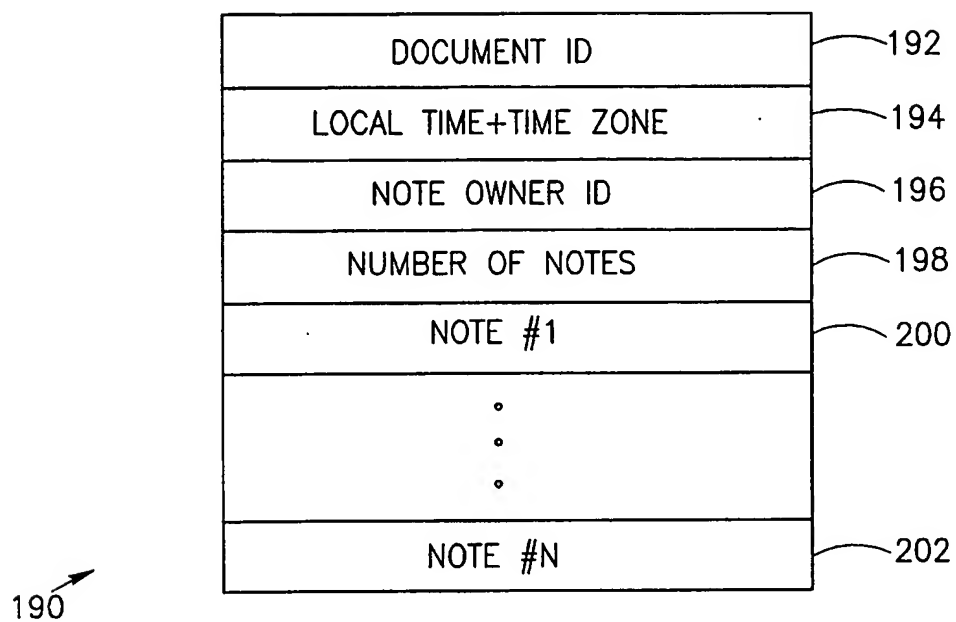


FIG.10

11/13

CLIENT ANNOTATION EVENT NOTE
DATA STRUCTURE

| | |
|--------------------------|-----|
| SERIAL NUMBER | 212 |
| ACTION | 214 |
| ANCHOR | 216 |
| STATUS | 218 |
| COLOR | 220 |
| TEXT LENGTH | 222 |
| TEXT | 224 |
| SHAPE | 226 |
| SHAPE LINE CAP | 228 |
| SHAPE LINE WIDTH | 230 |
| MARKER TYPE | 232 |
| NUMBER OF SHAPE VERTICES | 234 |
| SHAPE VERTICES | 236 |

210 ↗

FIG.11

12/13

SERVER ANNOTATION RESPONSE DATA STRUCTURE

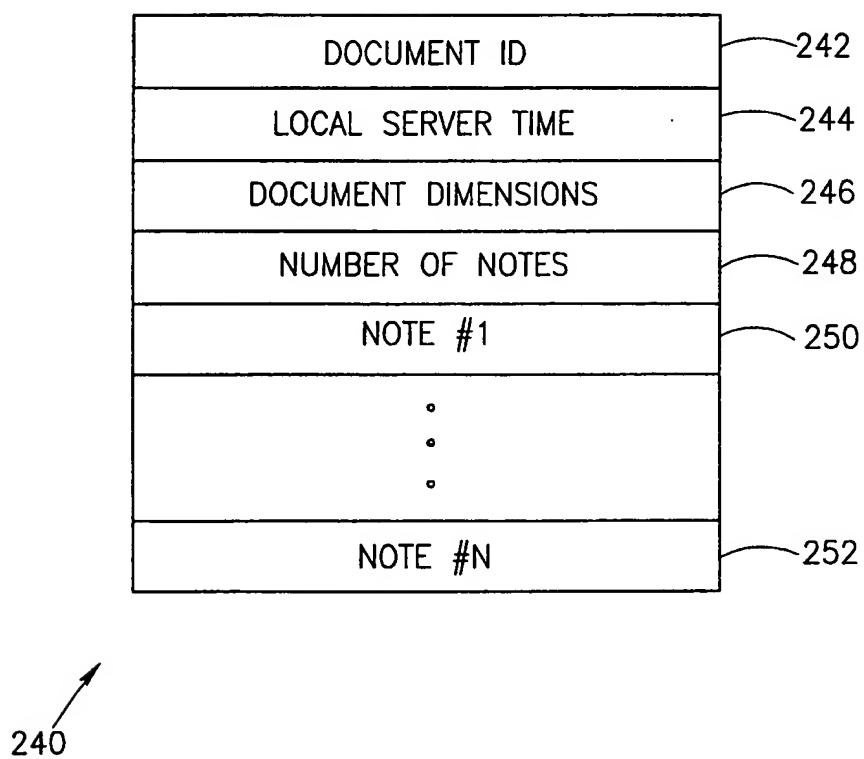


FIG.12

SERVER ANNOTATION RESPONSE
NOTE DATA STRUCTURE

| | |
|--------------------------|-----|
| SERIAL NUMBER | 262 |
| NOTE TIME STAMP | 264 |
| ACTION | 266 |
| ANCHOR | 268 |
| STATUS | 270 |
| COLOR | 272 |
| TEXT LENGTH | 274 |
| TEXT | 276 |
| SHAPE | 278 |
| SHAPE LINE CAP | 280 |
| SHAPE LINE WIDTH | 282 |
| MARKER TYPE | 284 |
| NUMBER OF SHAPE VERTICES | 286 |
| SHAPE VERTICES | 288 |

260 ↗

FIG.13